

1. A method of modeling a logic design, comprising:  
creating a graphical representation of the logic  
design; and  
generating simulation code based on the graphical  
5 representation.

2. The method of claim 1, wherein the graphical  
representation is comprised of functional block diagrams and  
virtual wires that interconnect the functional block  
10 diagrams.

3. The method of claim 2, wherein creating comprises:  
retrieving the functional block diagrams from a  
database; and  
15 arranging the functional block diagrams and the virtual  
wires to model the logic design.

4. The method of claim 2, wherein creating comprises:  
defining the functional block diagrams using simulation  
20 code; and  
arranging the functional block diagrams and the virtual  
wires to model the logic design.

5. The method of claim 1, further comprising:  
displaying a menu comprised of different types of  
functional block diagrams;  
receiving an input selecting one of the different types  
5 of functional block diagrams;  
retrieving a selected functional block diagram; and  
creating the graphical representation of the logic  
design using the selected functional block diagram.

10 6. The method of claim 1, further comprising:  
propagating a state through the simulation code; and  
determining if there is an error in the logic design  
based on the propagated state.

15 7. The method of claim 6, wherein the state comprises  
one of a zero state, a one state, and an undefined state.

20 8. The method of claim 6, further comprising:  
providing a visual indication if there is an error in  
the graphical representation of the logic design.

9. A method comprising:  
displaying a menu comprised of different types of  
functional block diagrams;  
receiving an input selecting one of the different types  
5 of functional block diagrams;  
retrieving a selected functional block diagram; and  
creating a graphical representation of a logic design  
using the selected functional block diagram;  
wherein creating comprises:

interconnecting the selected functional block  
10 diagram with one or more other functional block diagrams  
to generate a model of a logic design; and

defining the selected functional block diagram  
using simulation code if a function of the functional  
15 block diagram is undefined when retrieved.

10. The method of claim 9, further comprising:  
generating simulation code to simulate the operation of  
the logic design; and  
20 testing the logic design by propagating one or more  
states through the simulation code.

11. An article comprising a machine-readable medium that stores executable instructions for modeling a logic design, the instructions causing a machine to:

create a graphical representation of the logic design; and

generate simulation code based on the graphical representation.

12. The article of claim 11, wherein the graphical representation is comprised of functional block diagrams and virtual wires that interconnect the functional block diagrams.

13. The article of claim 12, wherein creating comprises:

retrieving the functional block diagrams from a database; and

arranging the functional block diagrams and the virtual wires to model the logic design.

14. The article of claim 12, wherein creating comprises:

defining the functional block diagrams using simulation code; and

arranging the functional block diagrams and the virtual wires to model the logic design.

5

15. The article of claim 11, further comprising instructions that cause the machine to:

display a menu comprised of different types of functional block diagrams;

receive an input selecting one of the different types of functional block diagrams;

retrieve a selected functional block diagram; and

create the graphical representation of the logic design using the selected functional block diagram.

16. The article of claim 11, further comprising instructions that cause the machine to:

propagate a state through the simulation code; and

determine if there is an error in the logic design based on the propagated state.

17. The article of claim 16, wherein the state comprises one of a zero state, a one state, and an undefined state.

5 18. The article of claim 16, further comprising instructions that cause the machine to:

provide a visual indication if there is an error in the graphical representation of the logic design.

10 19. An article comprising a machine-readable medium that stores executable instructions that cause a machine to:

display a menu comprised of different types of functional block diagrams;

15 receive an input selecting one of the different types of functional block diagrams;

retrieve a selected functional block diagram; and  
create a graphical representation of a logic design using the selected functional block diagram;

wherein creating comprises:

20 interconnecting the selected functional block diagram with one or more other functional block diagrams to generate a model of a logic design; and

defining the selected functional block diagram  
using simulation code if a function of the functional  
block diagram is undefined when retrieved.

5           20. The article of claim 19, further comprising  
instructions that cause the machine to:  
generate simulation code to simulate the operation of  
the logic design; and  
test the logic design by propagating one or more states  
through the simulation code.

20710 62450070  
20           21. An apparatus for modeling a logic design,  
comprising:  
a memory that stores executable instructions; and  
a processor that executes the instructions to:  
create a graphical representation of the logic  
design; and  
generate simulation code based on the graphical  
representation.

22. The apparatus of claim 21, wherein the graphical  
representation is comprised of functional block diagrams and

virtual wires that interconnect the functional block diagrams.

23. The apparatus of claim 22, wherein creating  
5 comprises:

retrieving the functional block diagrams from a database; and

arranging the functional block diagrams and the virtual wires to model the logic design.

24. The apparatus of claim 22, wherein creating  
comprises:

defining the functional block diagrams using simulation code; and

arranging the functional block diagrams and the virtual wires to model the logic design.

25. The apparatus of claim 21, wherein the processor executes instructions to:

20 display a menu comprised of different types of functional block diagrams;

receive an input selecting one of the different types of functional block diagrams;



retrieve a selected functional block diagram; and  
create the graphical representation of the logic design  
using the selected functional block diagram.

5           26. The apparatus of claim 21, wherein the processor  
executes instructions to:  
propagate a state through the simulation code; and  
determine if there is an error in the logic design based  
on the propagated state.

27. The apparatus of claim 26, wherein the state  
comprises one of a zero state, a one state, and an undefined  
state.

20           28. The apparatus of claim 26, wherein the processor  
executes instructions to:

provide a visual indication if there is an error in the  
graphical representation of the logic design.

20           29. An apparatus comprising:  
a memory that stores executable instructions; and  
a processor that executes the instructions to:

display a menu comprised of different types of  
functional block diagrams;

receive an input selecting one of the different  
types of functional block diagrams;

5           retrieve a selected functional block diagram; and  
          create a graphical representation of a logic design  
using the selected functional block diagram;  
          wherein creating comprises:

          interconnecting the selected functional block  
          diagram with one or more other functional block  
          diagrams to generate a model of a logic design; and

          defining the selected functional block diagram  
          using simulation code if a function of the  
          functional block diagram is undefined when  
          retrieved.

30. The apparatus of claim 19, wherein the processor  
executes instructions to:

20           generate simulation code to simulate the operation of  
the logic design; and

          test the logic design by propagating one or more states  
through the simulation code.